Overcoming the Switching Bottlenecks in Wavelength-Routing, Multicast-Enabled Architectures

Overcoming the Switching Bottlenecks in Wavelength-Routing, Multicast-Enabled Architectures

Kamran Keykhosravi, Houman Rastegarfar, Nasser Peyghambarian, and Erik Agrell

***Abstract*—Modular optical switch architectures combining wavelength routing based on arrayed waveguide grating (AWG) devices and multicasting based on star couplers hold promise for flexibly addressing the exponentially growing traffic demands in a cost- and power-efficient fashion. In a default switching scenario, an input port of the AWG is connected to an output port via a single wavelength. This can severely limit the capacity between broadcast domains, resulting in interdomain traffic switching bottlenecks. An unexplored solution to this issue is to exploit multiple AWG free spectral ranges (FSRs), i.e., to set up multiple parallel connections between each pair of broadcast domains. In this paper we *i)* study, for the first time, the influence of the FSR count on the throughput of a multistage switching architecture and *ii)* propose a generic and novel analytical framework to estimate the blocking probability. We assess the accuracy of our analytical results via Monte Carlo simulations. Our study points to significant improvements with a moderate increase in the number of FSRs. We show that an FSR count beyond four results in diminishing returns. Furthermore, to investigate the trade-offs between the network- and physical-layer effects, we conduct a cross-layer analysis, taking into account pulse amplitude modulation (PAM) and rate-adaptive forward error correction (FEC). We illustrate how the effective bit rate per port increases with an increase in the number of FSRs.**

***Index Terms*—Arrayed waveguide grating (AWG), blocking probability, coupler, free spectral range (FSR), multicast, physical layer, scheduling, switch architecture.**

I. Introduction

With the proliferation of smart mobile devices, the continuous advances in computational power, and the breakthroughs in the field of machine learning, the fifth generation of cellular networks (5G) is being rolled out to provide dramatic improvements in the throughput, latency, and reliability performance for a myriad of services and applications [1]–[5]. With new technologies such as the Internet of Things (IoT), high-resolution video streaming, road safety, wearable devices, and augmented reality, and with increasing capacity demands for large-scale scientific calculations, the network traffic is growing at an exponential pace across all geographical spans. To cope with the ever-increasing traffic rates in a sustainable fashion, innovative and intelligent networking solutions that simultaneously optimize the transmission, architecture, and control and management aspects have become indispensable [6]–[8].

The abundant capacity and power efficiency of wavelength-division multiplexed (WDM) networks makes them a promising candidate for interconnecting computing nodes in a data center environment and wireless endpoints in a 5G networking scenario. Optical interconnect designs that support the launch of tens of wavelengths per fiber provide for ultrahigh switching capacities, bit-rate transparency, low power density, resource virtualization flexibility, and acceleration in the execution of large-scale distributed applications. Due to their compelling properties, wavelength-routing interconnects, based on the cyclic routing pattern of arrayed waveguide grating (AWG) as a passive and low-footprint switching device, have received attention for use in both large-scale data centers [8]–[10] and the fronthaul segment of radio access networks [11].

While the switching potential of the AWG is limited by its static point-to-point routing pattern (in an $N \times N$ AWG with N available wavelengths, each input port is connected to each output port with a fixed wavelength), AWG-based switch architectures can be made highly flexible by incorporating optical components with complementary switching capabilities [8], [12], [13]. For instance, star couplers enable nonblocking unicast, multicast, and broadcast traffic delivery directly in the optical domain and can be added to a wavelength-routing architecture to support a rich set of traffic patterns. Multicast traffic involves the simultaneous dissemination of the same information copy to a group of recipients and constitutes a major portion of data center traffic (e.g., MapReduce) [14]–[21]. As well, advanced coordinated multipoint (CoMP) transmission techniques in radio access networks call for efficient optical multicasting from a central office (node) to a group of cooperating radio heads [22], [23].

Recently, a hierarchical, wavelength-routing switch architecture with distributed broadcast domains has been proposed for scalable and flexible optical switching in data centers [8]. As depicted in Fig. 1, this design interconnects a maximum of $N \times (K-1)$ nodes using an $N \times N$ AWG

K. Keykhosravi and E. Agrell are with the Department of Electrical Engineering, Chalmers University of Technology, Gothenburg 412 96, Sweden (e-mail: kamrank@chalmers.se; agrell@chalmers.se).

H. Rastegarfar and N. Peyghambarian are with the College of Optical Sciences, University of Arizona, Tucson, Arizona 85721, USA (e-mail: houman@optics.arizona.edu; nasser@optics.arizona.edu).

Manuscript received August 14, 2018; revised August 14, 2018.

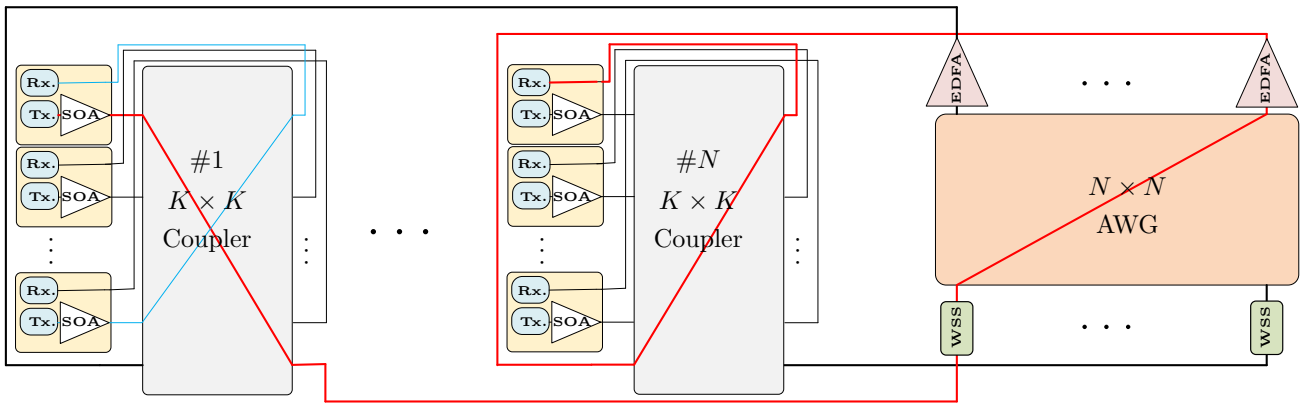

Fig. 1. A distributed multicast architecture based on star couplers and AWG [8]. The signal path for an intradomain connection (in thin blue line) and an interdomain connection (in thick red line) are shown. SOA: semiconductor optical amplifier. EDFA: erbium-doped fiber amplifier. WSS: wavelength-selective switch.

and N $K \times K$ star couplers. The architecture in Fig. 1 allows the same set of wavelengths to be used in each broadcast domain (hence, named the wavelength-reuse architecture) in order to overcome the scalability issues due to the limited coupler port count and transceiver tuning range. Ideally, the nodes attached to different couplers should be able to freely communicate with each other. However, two different factors undermine the scalability of the wavelength-reuse architecture. First, with the number of wavelengths and AWG ports being equal, a given pair of couplers can only use one wavelength to communicate with each other. Second, the physical-layer impairments that accumulate over multiple routing stages can severely limit the achievable capacity per such a wavelength [24].

In order to overcome the above-mentioned segmentation mechanisms, it is crucial to improve the interdomain (i.e., intercoupler) switching capacity as much as possible through innovations at both hardware and scheduling levels. Our proposed solution is to utilize the free spectral range (FSR) periodicity of AWG. In an $N \times N$ AWG with an FSR count of F, each input port can be connected to each output port via F distinct wavelengths [25]–[27]. Assuming that the range of available wavelengths is fixed, employing a larger F results in a smaller port count. Hence, there is trade-off between the number of supported nodes and the blocking probability (BP) of the design.

Contributions: The contributions of this paper are as follows: *i)* We examine the impact of AWG FSR periodicity on the performance of the wavelength-routing switch in Fig. 1. We quantify performance in terms of blocking probability, bit error rate (BER), and throughput. To the best our knowledge, this is the first work to thoroughly investigate the FSR count impact on the performance of wavelength-routing switches. *ii)* We propose a novel analytical framework for estimating the BP of the switch. The estimation accuracy is confirmed via Monte Carlo simulations. Although our analysis is performed under a specific switching scenario, it can be adapted to estimate the BP of any generic multistage switch with interconnected broadcast domains. Our results point to significant performance gains of exploiting AWG FSRs. Furthermore, it is shown that the switch performance saturates at $F = 4$. By conducting physical-layer simulations for PAM of orders 2, 4, and 8, we show that 4-PAM is the most suitable modulation order for multi-FSR interdomain communication.

The remainder of this paper is organized as follows. In Section II, we present the scheduling algorithm used in the paper, which is a generalization of the scheduler in [8] to the multi-FSR case. In Section III, we derive an analytical approximation of the BPs in the distributed broadcast architecture. In Section IV, we study the impact of FSR periodicity via Monte Carlo simulations and compare the simulation results with the analytical results obtained in Section III. Section V is devoted to studying the impact of the physical layer impairments in a multi-FSR switching scenario. We report BER and normalized throughput values, considering various modulation orders, FSR counts, and adaptive coding. Finally, Section VI summarizes and concludes the paper.

II. A Multi-FSR Scheduling Algorithm

In this section, we describe a multi-FSR scheduling algorithm, which is a generalization of the algorithm in [8], for the architecture of Fig. 1. We assume the AWG port count is N and that the number of available wavelengths is N_W. Hence, the number of available FSRs is $F = N_W/N$. The value of F indicates the number of wavelengths that connect an arbitrary AWG input port to each output port. Specifically, the ith AWG input port can be connected to the jth AWG output port via wavelengths

$$(f-1)N + \mathrm{mod}\,(i+j-1, N), \quad f = 1, \ldots, F. \tag{1}$$

Table I illustrates the input–output connection map of a 4×4 AWG with $F = 4$ ($N_W = 16$) as an example.

TABLE I
ROUTING PATTERN OF A 4×4 AWG WITH $F = 4$. EACH INPUT PORT IS CONNECTED TO EACH OUTPUT PORT VIA 4 WAVELENGTHS.

In \ Out	1	2	3	4
1	$\lambda_1, \lambda_5,$ λ_9, λ_{13}	$\lambda_2, \lambda_6,$ $\lambda_{10}, \lambda_{14}$	$\lambda_3, \lambda_7,$ $\lambda_{11}, \lambda_{15}$	$\lambda_0, \lambda_4,$ λ_8, λ_{12}
2	$\lambda_2, \lambda_6,$ $\lambda_{10}, \lambda_{14}$	$\lambda_3, \lambda_7,$ $\lambda_{11}, \lambda_{15}$	$\lambda_0, \lambda_4,$ λ_8, λ_{12}	$\lambda_1, \lambda_5,$ λ_9, λ_{13}
3	$\lambda_3, \lambda_7,$ $\lambda_{11}, \lambda_{15}$	$\lambda_0, \lambda_8,$ $\lambda_{12}, \lambda_{16}$	$\lambda_1, \lambda_5,$ λ_9, λ_{13}	$\lambda_2, \lambda_6,$ $\lambda_{10}, \lambda_{14}$
4	$\lambda_0, \lambda_4,$ $\lambda_8, \lambda_{12},$	$\lambda_1, \lambda_5,$ λ_9, λ_{13}	$\lambda_2, \lambda_6,$ $\lambda_{10}, \lambda_{14}$	$\lambda_3, \lambda_7,$ $\lambda_{11}, \lambda_{15}$

Let $L_{i,j}$ denote the link from input port i of the AWG to its output port j, where $1 \leq i, j \leq N$. Consider two links $L_{i,j}$ and $L_{j,i}$, where $i \neq j$. Due to the reciprocal property of the AWG, the same set of wavelengths can be used to transmit through $L_{i,j}$ and $L_{j,i}$. If a wavelength is simultaneously used in both links, collision will occur in couplers i and j. For example in the setting described by Table I, assume that a connection is established from coupler 1 to coupler 2 using λ_2, and at the same time another one is established from coupler 2 to coupler 1 with the same wavelength. The signal sent by the transmitter in coupler 1 is routed through the AWG to coupler 2, where it interferes with the signal sent by the transmitter in coupler 2 since they have the same wavelength (λ_2). Therefore, in order to prevent collisions, the scheduling algorithm should take into account the appropriate allocation of wavelengths. In short, a specific wavelength can only be used either in $L_{j,i}$ or in $L_{i,j}$ but not in both. As a result, with $F = 1$, only one of $L_{i,j}$ and $L_{j,i}$ can be allowed to transmit data. With larger F values, however, the available wavelengths can be split and allocated fairly between the two links.

We define some notations that are used in our scheduling algorithm. Let $W_{i,j}$ be the set of all wavelengths that can be used to transmit through $L_{i,j}$. As an example, in the AWG represented in Table I, $W_{1,2} = \{\lambda_2, \lambda_6, \lambda_{10}, \lambda_{14}\}$. Moreover, we define two subsets $W^1_{i,j}$ and $W^2_{i,j}$ that have the same cardinality and partition the set $W_{i,j}$ into two equivalent sets of wavelength resources, e.g., $W^1_{1,2} = \{\lambda_2, \lambda_6\}$ and $W^2_{1,2} = \{\lambda_{10}, \lambda_{14}\}$[1]. The reason for partitioning the wavelength set $W_{i,j}$ into two subsets is to improve the fairness of the scheduling algorithm by evenly distributing the available wavelengths between links $L_{i,j}$ and $L_{j,i}$.

The multi-FSR scheduling algorithm consists of two phases. First, all the interdomain traffic (i.e., connections whose source and destination nodes reside in different couplers) is scheduled and next the intradomain traffic (i.e., connections whose source and destination nodes reside in the same coupler). The scheduling steps for the first phase are as follows.

1) Begin by considering the traffic requests to all destination nodes in coupler d. For fairness, the starting point $1 \leq d \leq N$ is chosen randomly or using a round-robin pointer that is updated in each scheduling cycle.
2) Choose randomly one of the destination nodes in the dth coupler with minimum (and non-zero) number of requests.
3) Select randomly a source node in another coupler s requesting that destination.
4) If $s > d$, pick one available wavelength in $W^1_{s,d}$ randomly to schedule the connection and mark this wavelength as unavailable from coupler d. Block the request if no available wavelength exists. If $s < d$, use $W^2_{s,d}$ instead of $W^1_{s,d}$.
5) Repeat steps 2, 3, and 4 until all requests to coupler d are granted or blocked. Afterwards, schedule the interdomain requests destined to other couplers than d in the same fashion.
6) Restore all the blocked requests. To take advantage of any remaining wavelength resources, perform all the previous steps except Step 4, which is replaced with Step 4* as follows.

4*) If there exists an available wavelength in $W_{s,d}$, use it to schedule the request and mark it as unavailable in coupler d. Otherwise, block it.

7) If all couplers d have been examined, terminate the first scheduling phase and go to the second one. Otherwise, update d and go to Step 2.

With $F = 1$, Step 4 is simply replaced with Step 4* and Step 6 is removed[2]. In this case, the algorithm reduces to the one proposed in [8, Sec. II-B]. In Steps 1 to 5, the scheduling is performed by fairly dividing the available wavelengths into two disjoint sets for transmission from coupler i to coupler j or vice versa. In Step 6, to ensure work conserving property (see [20, Sec. III-C]), the connections are scheduled using all of the available wavelengths in each link. Furthermore, in Step 2 the priority is given to destination nodes with the minimum number of requests to minimize the BP [8].

After scheduling the interdomain traffic, the second phase of the algorithm is carried out to schedule the intradomain traffic. This phase is identical to the algorithm in [8]; however, we present it here for the sake of completeness. According to Fig. 1, with $K \times K$ star couplers, each coupler is directly attached to $K-1$ source and destination nodes. The following intradomain traffic scheduling tasks are carried out in each coupler in parallel.

1) Begin the scheduling from a destination node $1 \leq o \leq K-1$, where o is chosen randomly or according to a round-robin pointer updated in each scheduling cycle.
2) If either the destination node has already been matched or no intradomain traffic is destined to that node, go to Step 4. Otherwise, randomly select one of the sources requesting that destination.

[1]Here, we assume that F is an even number.

[2]Note that with $F = 1$ the cardinality of $W_{s,d}$ is equal to one and it cannot be split into two subsets.

3) Schedule the connection via a wavelength that has not been used in the coupler. The wavelength can be picked randomly or based on a first-fit wavelength assignment policy. If all wavelengths are occupied, block the request and terminate the scheduling.
4) Repeat Step 1 for updating o and Steps 2 and 3 until all of the destinations have been examined.

The complexity of the scheduling algorithm is $\mathcal{O}(KN)$ since the scheduler goes over all destination nodes. The algorithm is simple since there is no complicated mathematical operation involved in the scheduling process. Furthermore, the algorithm is distributed and can be implemented in parallel. Hence, the presented scheduler is scalable.

As has been mentioned at the beginning of this section, our algorithm is a generalization of the one proposed in [8]. This is because *i)* it reduces to the algorithm in [8] for $F = 1$ and *ii)* it enables fair scheduling of the connections for any F. To see the latter point, assume that $F = 2$ and two connections from Coupler 1 to Coupler 2 and two from Coupler 2 to Coupler 1 exist. The algorithm proposed in [8] schedules only the two connections from Coupler 1 to Coupler 2 and blocks the other two (assuming that it begins the scheduling from Coupler 1). However, with our algorithm one connection from each coupler is scheduled and one is blocked, which is a fair resource allocation approach.

III. Blocking Probability Analysis

In this section, we present an analytical framework for estimating the (interdomain and intradomain) blocking probabilities of the distributed multicast architecture in Fig. 1. Here, the BP is defined as the probability that a single connection request is blocked by the scheduler in each scheduling cycle. This section only considers the BP caused by output-port contentions. A connection can also be blocked due to poor signal quality; however, we take this into consideration when calculating the switch throughput in Section V. We consider optical circuit switching with offline scheduling, i.e., we assume that at each scheduling instance, an input port of the switch has a connection request with probability ρ independently of other instances. The parameter ρ can be regarded as average input port utilization or the normalized load of the switch. Furthermore, we let $0 \leq R_{\text{inter}} \leq 1$ be the probability that a connection is destined to a node outside its corresponding broadcast domain. The destination of each interdomain (intradomain) connection is chosen uniformly among all of the nonlocal (local) nodes.

To conduct our analytical study, firstly, we derive the BP of a single star coupler (i.e., a strict-sense nonblocking switch) in Section III-A. Secondly, in Sections III-B and III-C, we calculate the approximate BP of the interdomain traffic for $F = 1$ and $F = 2$, respectively. We investigate larger values of F in Section III-D. Thirdly, in Section III-E we derive the intradomain BP.

A. Blocking probability in a star coupler

Assume that a switch consisting of a single star coupler simultaneously receives K_{in} connection requests from distinct input ports. Each request is destined to a port that is chosen randomly, independently, and uniformly among K_{out} output ports. If multiple requests involve the same output port, one of these requests, chosen randomly and uniformly, is accepted, while all other requests for that output port are blocked. The BP is in this scenario given by the following lemma.

Lemma 1. Each of the K_{in} connection requests is blocked with probability

$$\text{BP}(K_{\text{in}}, K_{\text{out}}) = 1 - \frac{K_{\text{out}} - \text{E}[\mathbf{n}_{\text{idle}}]}{K_{\text{in}}}. \tag{2}$$

Here, the random variable $\mathbf{n}_{\text{idle}}$ is the number of idle output ports, whose average is

$$\text{E}[\mathbf{n}_{\text{idle}}] = K_{\text{out}} \left(1 - \frac{1}{K_{\text{out}}}\right)^{K_{\text{in}}}. \tag{3}$$

Proof: The number of blocked connections is $K_{\text{in}} - \mathbf{n}_{\text{busy}}$, where $\mathbf{n}_{\text{busy}}$ denotes the number of busy output ports, that is, $K_{\text{out}} - \mathbf{n}_{\text{idle}}$. Therefore, the BP can be calculated as $\text{E}[K_{\text{in}} - (K_{\text{out}} - \mathbf{n}_{\text{idle}})]/K_{\text{in}}$, which is equal to (2). In the following, we calculate $\text{E}[\mathbf{n}_{\text{idle}}]$.

Define random variable $\mathbf{x}_i, 1 \leq i \leq K_{\text{out}}$ to be 1 if the ith output port is idle and 0 otherwise. The number of idle output ports can be expressed as

$$\mathbf{n}_{\text{idle}} = \sum_{i=1}^{K_{\text{out}}} \mathbf{x}_i. \tag{4}$$

Therefore, we have

$$\text{E}[\mathbf{n}_{\text{idle}}] = \sum_{i=1}^{K_{\text{out}}} \text{E}\left[\mathbf{x}_i\right]. \tag{5}$$

To calculate $\text{E}\left[\mathbf{x}_i\right]$, one should note that for all i, the ith output port is idle with probability

$$\Pr\left(\mathbf{x}_i = 1\right) = \left(1 - \frac{1}{K_{\text{out}}}\right)^{K_{\text{in}}}. \tag{6}$$

Hence, from (5) and (6), we have

$$\text{E}[\mathbf{n}_{\text{idle}}] = K_{\text{out}} \left(1 - \frac{1}{K_{\text{out}}}\right)^{K_{\text{in}}}. \tag{7}$$

■

We note that Lemma 1 can be cast into an occupancy problem, where K_{in} balls (connection requests) are tossed to K_{out} bins (output ports) (see for example [28, Ch. 2]).

Although Lemma 1 is formulated to calculate the BP in a star coupler, it can also be used in other nonblocking scenarios. We will use this lemma to derive the BPs of the switch in Fig. 1 in the subsequent sections.

B. Interdomain blocking probability with $F = 1$

We consider the case of interdomain routing in the switch of Fig. 1 with $F = 1$. We denote with $\mathcal{S}_{i,j}$ the set of all connections whose source and destination lie in couplers i and j, respectively. For tractable mathematical analysis, we consider a simplified version of the scheduling algorithm developed in the previous section. With this simplified scheduler, the interdomain connections are scheduled in three steps. First, one connection is chosen at random among each set $\mathcal{S}_{i,j}$, $1 \leq i, j \leq N$, and the rest of the connections are blocked. Second, if there exist a connection request from coupler i to coupler j and a connection request from coupler j to i, the scheduler randomly chooses one of them and blocks the other[3]. Then, each connection is assigned a wavelength based on the AWG routing pattern. Third, among all non-blocked connections destined to each receiver node, only one is chosen randomly and the others are blocked (to resolve output port contention). Each receiver is then tuned to the wavelength of its corresponding connection.

Please note that in the original scheduling algorithm (in Sec. II), the interdomain traffic is scheduled from the output side (i.e, per destination coupler). However, in this analysis we consider a three-stage scheduling, moving from the source to the AWG and then to the destination. Besides, we relax the work-conserving requirement of the scheduler. Let b_1, b_2, and b_3 be the BP due to the first, second, and third scheduling steps, respectively. The total BP can be calculated as

$$b_{\text{inter}} = 1 - (1 - b_1)(1 - b_2)(1 - b_3). \tag{8}$$

In each step of our calculations, we replace all random variables with their mean value to simplify the analysis. As a result, we assume that $(K-1)\rho$ connections are present in the input ports of each coupler, out of which

$$m_1 = R_{\text{inter}}(K-1)\rho \tag{9}$$

are interdomain connections. It is easy to notice that b_1 is actually equal to the BP of a coupler with m_1 input connections and a destination set of cardinality $N-1$. Using Lemma 1, we have

$$b_1 \approx \text{BP}(m_1, N-1). \tag{10}$$

To calculate b_2, we assume that

$$m_2 = m_1(1 - b_1) \tag{11}$$

connection requests are present on each input port of the AWG. In the second scheduling step, a connection request from coupler i to coupler j is blocked with probability $1/2$ should there exist a connection request from coupler j to coupler i, which happens with probability $m_2/(N-1)$. Therefore,

$$b_2 \approx \frac{m_2}{2(N-1)}. \tag{12}$$

[3]Note that setting up both connections leads to contention. See Sec. II.

Finally, to calculate b_3, we assume that a total number of $m_3 = Nm_2(1 - b_2)$ connections should be scheduled during the third step of the algorithm. The destination set of the switch has the cardinality of $N(K-1)$. Here, again the problem can be solved via Lemma 1. Note that the blocking properties of the switch architecture have already been taken into account in Steps 1 and 2. We obtain

$$b_3 \approx \text{BP}(m_3, N(K-1)). \tag{13}$$

In (13), for simplification, we neglect that an interdomain connection is not allowed to be destined to its source domain. This concludes the estimation of the interdomain BP under $F = 1$.

C. Interdomain blocking probability with $F = 2$

To calculate the BP with $F = 2$, we consider a simplified scheduler that performs the scheduling tasks in two iterations. Each iteration involves three steps similar to the ones in the scheduler presented in Sec. III-B. The scheduling steps are as follows.

1) Randomly choose one connection request from each set $\mathcal{S}_{i,j}$. We use $\tilde{\mathcal{S}}_{i,j}$ to denote the set of remaining connection requests.
2) Assign a wavelength to each of the chosen connections based on their destination. In this step, no connection is blocked as two wavelengths are available for setting up connections between coupler i and coupler j ($1 \leq i, j \leq N$).
3) For each receiver, one connection is chosen among all the ones destined to it and the rest are blocked.
4) One connection is randomly selected from each set $\tilde{\mathcal{S}}_{i,j}$ and the rest are blocked.
5) If available, a wavelength is assigned to each connection according to its destination. Otherwise, the connection is blocked.
6) For each free receiver node, one connection is selected out of all connections destined to it, and the rest are blocked. Moreover, the connections destined to busy receivers are blocked.

We represent the BP at step ℓ by $b_\ell, 1 \leq \ell \leq 6$. For the first step, b_1 can be approximated as in (10). We have $b_2 = 0$. Similarly to (13), $b_3 \approx \text{BP}(m_3, N(K-1))$. The average of the cardinality of set $\tilde{\mathcal{S}}_{i,j}$ is $b_1 m_1$, where m_1 is defined in (9). Therefore, similarly as in (10), $b_4 \approx \text{BP}(b_1 m_1, N-1)$. To approximate b_5, we only consider one (the most probable) event, where both wavelengths for transmission between couplers i and j have been used in the second step, one for transmission from coupler i to j and the other from j to i. As discussed in Sec. III-B, this probability can be approximated as $b_5 \approx m_2/(N-1)$, where m_2 is defined in (11). In the sixth step, a connection is blocked either if it is destined to a busy receiver or if it is in contention with other connections. The probability of the former event can be approximated by $b_6^{(1)} \approx m_4/(N(K-1))$, where $m_4 = Nm_1(1 - b_1)(1 - b_3)$

is the average number of busy receivers. The probability of the latter event can be calculated similarly as in (13) and is $b_6^{(2)} \approx \mathrm{BP}(m_5, N(K-1)-m_4)$, where $m_5 = Nb_1m_1(1-b_4)(1-b_5)(1-b_6^{(1)})$ is the average number of connections in Step 6 that are destined to free receivers. Knowing the BP of each step, we first calculate the total number of scheduled connections, T, as

$$T = m_1(1-b_1)(1-b_2)(1-b_3) + b_1m_1(1-b_4)(1-b_5)(1-b_6^{(1)})(1-b_6^{(2)}) \quad (14)$$

Finally, the total interdomain BP can be calculated as

$$b_{\text{inter}} = 1 - T/m_1. \quad (15)$$

D. Interdomain blocking probability with $F > 2$

In this section, we present an algorithm to approximate the interdomain blocking probability for FSR counts larger than 2. To do so, we pursue the same analysis as in Sections III-B and III-C. To make the analysis tractable, we neglect the blocking events that arise due to having two simultaneous connection requests from coupler i to j and from j to i, when there are not enough wavelengths available. When F is large, the probability of such events is negligible. However, with $F = 1$ and $F = 2$, this probability is considerable and the proposed algorithm results in a very optimistic approximation. For these two cases (8) and (15) should be used to approximate the BP.

The pseudocode of the proposed algorithm is presented in Algorithm 1. A simplified scheduler similar to the one in Section III-C is adopted, which performs the scheduling during F iterations. In each iteration, a connection from coupler i to coupler j can be blocked because: *i)* it loses the competition to other connections from i to j (shown by b_1 in Algorithm 1), *ii)* it is destined to a busy node (shown by b_2 in Algorithm 1), or *iii)* it loses the competition to other connections that have the same destination node (denoted by b_3 in Algorithm 1). m_1 represents the average number of connections per coupler at the beginning of each scheduling iteration. m_2 is the number of nonblocked connections destined to free receivers. T represents the number of scheduled connections. The calculation of these parameters in Algorithm 1 is performed similarly as is in Section III-C.[4]

E. Intradomain blocking probability

After approximating the BP of the interdomain traffic, one can invoke Lemma 1 to evaluate that of the intradomain traffic. The average number of busy receivers per coupler, after scheduling the interdomain traffic, can be approximated by

$$n_b \approx m_1(1-b_{\text{inter}}). \quad (16)$$

where m_1 is defined in (9). The average number of free receivers is $n_f = K-1-n_b$. The intradomain BP consists of two factors: *i)* the probability that a connection is destined to a busy receiver, denoted by $\tilde{b}_1$ and *ii)* the probability of contention among the connections destined to a free receiver, denoted by $\tilde{b}_2$. We have that $\tilde{b}_1 \approx n_b/(K-1)$. Moreover, by Lemma 1 we have

$$\tilde{b}_2 \approx \mathrm{BP}\left((1-R_{\text{inter}})(1-\tilde{b}_1)(K-1)\rho, n_f\right). \quad (17)$$

Thus, the interdomain BP can be calculated as

$$b_{\text{intra}} = 1-(1-\tilde{b}_1)(1-\tilde{b}_2). \quad (18)$$

Algorithm 1

Inputs: N: AWG port count; K: Coupler port count; ρ: Average input port utilization; R_{inter}: Probability of interdomain connection; F: FSR value.
Output: Blocking probability of the switch, b_{inter}.

$T \leftarrow 0$
$m_1 \leftarrow R_{\text{inter}}(K-1)\rho$
for *counter* $\in \{1,\dots,F\}$ **do**
  $b_1 \leftarrow \mathrm{BP}(m_1, N-1)$
  $b_2 \leftarrow T/(K-1)$
  $m_2 \leftarrow Nm_1(1-b_1)(1-b_2)$
  $b_3 \leftarrow \mathrm{BP}\left(m_2, N(K-1)-NT\right)$
  $T \leftarrow T + m_1(1-b_1)(1-b_2)(1-b_3)$
  $m_1 \leftarrow m_1 \cdot b_1$
return $b_{\text{inter}} = 1 - T/\left(R_{\text{inter}}(K-1)\rho\right)$

IV. Numerical validation

In this section, we evaluate the blocking probabilities of the distributed multicast architecture via Monte Carlo simulations for $F \in \{1,2,3,4\}$ and compare them with the analytical approximations derived in Section III-B for $F = 1$, Section III-C for $F = 2$, and Section III-D (Algorithm 1) for $F = 3$ and $F = 4$. It is assumed that the number of available wavelengths is fixed and equals $N_W = 64$. As a result, the AWG port count scales as $N = 64/F$. Each reported value corresponds to the average over $10,000$ simulation runs. Throughout the paper, we set $R_{\text{inter}} = 0.25$.

Figure 2 represents the interdomain BP versus load. Along with the simulation results, the analytical BP values for are plotted in Fig. 2. As can be seen, the analytical results are in close agreement with the simulation results. The small differences are mainly due to two simplifications that we made in our analysis. First, instead of considering the distribution of the random variables, we considered their expected value, in each step. Second, we analyzed a simplified scheduling algorithm, while the one presented in Sec. II was simulated.

An interdomain connection is blocked for two reasons, i.e., wavelength shortage in the AWG or output port contention. The former is more significant for small values of F. With an increase in F, more wavelengths become available to connect AWG input and output ports. The BP due to wavelength shortage can be significantly reduced with a proper choice of FSR count. According to Fig. 2, increasing F past 4 has diminishing returns.

[4] Specifically, in Algorithm 1 b_1 is calculated similarly as b_1 or b_3 in Section III-C; b_2 as $b_6^{(1)}$; m_2 as m_5; b_3 as b_3 or $b_6^{(2)}$; T as T; and m_1 as m_1.

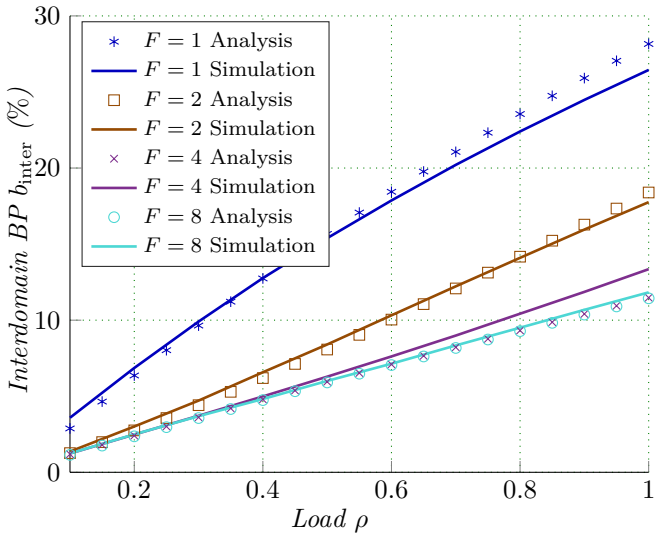

Fig. 2. Simulation and analytical results for interdomain BP b_{inter} of the switch in Fig. 1 for $F \in \{1, 2, 4, 8\}$.

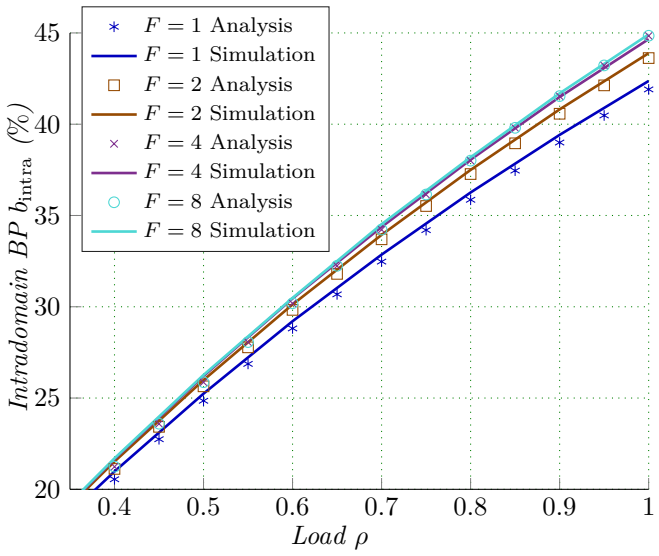

Fig. 3. Simulation and analytical results for intradomain BP b_{intra} of the switch in Fig. 1 for $F \in \{1, 2, 4, 8\}$.

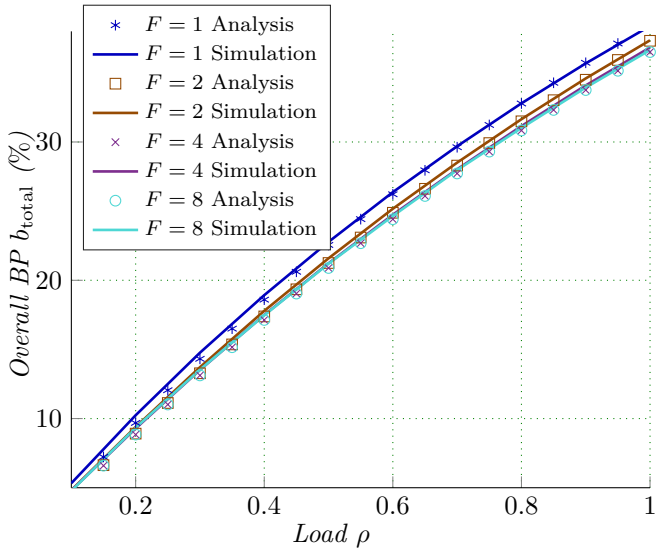

Fig. 4. Simulation and analytical results for overall BP b_{total} of the switch in Fig. 1 for $F \in \{1, 2, 4, 8\}$.

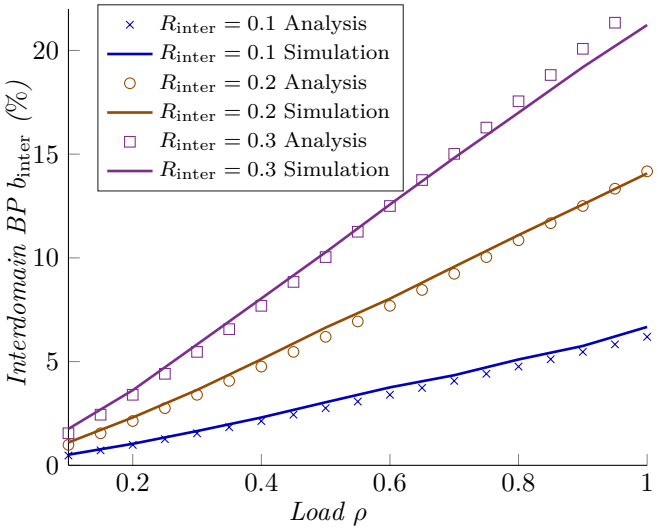

Fig. 5. Simulation and analytical results for interdomain BP b_{inter} of the switch in Fig. 1 for $F = 2$ and $R_{\text{inter}} \in \{0.1, 0.2, 0.3\}$.

Figure 3 depicts the simulation and analytical results in terms of intradomain BP for $F = 1, 2, 4, 8$. As in the case of interdomain traffic, a good agreement exists between the simulation and analytical results. As depicted in Fig. 2, an increase in F results in lower interdomain blocking probabilities; hence, a larger portion of receivers become occupied by the interdomain traffic. This explains why the intradomain BP degrades with an increase in F. Besides, for a given load, the intradomain traffic suffers a higher BP compared with the interdomain traffic. This is primarily due to the fact that the multi-FSR scheduler prioritizes the interdomain connections by trying to allocate them resources first.

Fig. 4 illustrates the overall BP based on simulation and analysis, where the analytical approximation is

$$b_{\text{total}} = R_{\text{inter}} b_{\text{inter}} + (1 - R_{\text{inter}}) b_{\text{intra}}. \tag{19}$$

As can be seen, (19) approximates the BP with a high accuracy. Based on (19), the overall BP can be written as a weighted average of interdomain and intradomain BPs. The former decreases with F (see Fig. 2) while the latter increases with F (see Fig. 3). Fig. 4 shows that the overall BP follows the trend of interdomain BP as it becomes smaller when F grows larger.

Finally, in Fig. 5, we investigate the impact of traffic locality on the BP for $F = 2$ by considering three different values of $R_{\text{inter}} \in \{0.1, 0.2, 0.3\}$. By increasing R_{inter}, the interdomain traffic rate increases, which leads

TABLE II
Simulation parameters

Parameter	Value
Number of simulation runs	$10,000$
FSR count (F)	$1, 2, 4, 8$
Coupler port count (K)	64
Wavelength count (N_W)	64
AWG port count (N)	$64/F$
Symbol rate	28 Gbaud
R_{inter}	0.25

to higher BP as is evident in Fig. 5. We can observe a close agreement between our analytical approximation and simulation results.

For the sake of tractable mathematical analysis, all results presented in this section are based on uniform traffic load. That is, we assume that in each scheduling instance nodes generate connection request with the same probability ρ. However, connection requests do not necessarily follow a uniform generation process in a realistic setting. We have also examined the switch performance in a nonuniform traffic generation scenario (in which half of the nodes have twice the load of the other half), and have observed that our theoretical model may also be applied to more general traffic patterns. A comprehensive study of the BP under nonuniform traffic is an interesting topic for future studies.

V. Cross-Layer Performance Analysis

A signal traversing the different routing stages of the distributed multicast architecture is affected by multiple impairments, namely thermal noise, laser relative intensity noise, shot noise, amplified spontaneous emission (ASE) noise, in-band AWG crosstalk, and out-of-band crosstalk. Therefore, the transmitted symbols are detected at the receiver with errors. Fig. 6 illustrates the signal path from the transmitter to its destination for interdomain and intradomain traffic. The ASE noises of the amplifiers are shown at the output of the amplification stages. A comprehensive physical-layer model for pulse amplitude modulation (PAM) has been developed in [8]. We use the same analytical setup with the same physical-layer parameters (see [8, Table I]) to model the signal propagation and calculate the BER. The physical-layer parameters are selected based on datasheets for commercially available products and recent lab demonstrations [29]–[33]. As in [8], we conduct Monte Carlo simulations to evaluate the effects of crosstalk on the signal. Some of the key simulation parameters are presented in Table II.

Figure 7 illustrates the (overall) BER versus load for three modulation orders and $F = 1, 2, 4, 8$. We can distinguish three trends in Fig. 7. First, an increase in F results in a decrease in BER. This is due to an increase in the in-band crosstalk in the AWG. With a smaller value of F, we have an AWG with a larger port count. Therefore, for an arbitrary interdomain connection, on the average, the number of connections that traverse the AWG with the same wavelength is higher, which translates to a higher in-band crosstalk. Second, the BER monotonically increases with load, which is due to the intensified crosstalk originating from more co-propagating channels. Third, increasing the order of the modulation increases the BER. With 8-PAM, the BER can exceed 10^{-2} while with 2-PAM (i.e., on–off keying), the transmission is virtually error-free.

To investigate the impact of physical-layer impairments on the transmission rate for different modulation orders, we deploy a forward error correction (FEC) code with rate adaptation. We use a Reed–Solomon code with block length of 255 bytes [34], i.e., RS$(255, k)$, where k is chosen by the switch controller after calculating the pre-FEC BER such that the post-FEC BER becomes less than 10^{-12}. The larger the pre-FEC BER, the smaller the value of k, and the lower the effective bit rate per transmitter. We assume that if the pre-FEC BER is larger than $3{\cdot}10^{-2}$, the signal cannot be retrieved at the receiver. We focus on the interdomain traffic as it is more susceptible to the impairments than the intradomain traffic.

Figure 8 depicts $\mathcal{T}_{\text{inter}}$, that is the average interdomain throughput per node normalized by R_{inter}, versus load for $F \in \{1, 2, 4, 8\}$. Specifically, the vertical axis in Fig. 8 represents

$$\mathcal{T}_{\text{inter}} = \frac{\text{Average total interdomain traffic}}{N(K-1) \times R_{\text{inter}}}. \tag{20}$$

For each modulation order, $\mathcal{T}_{\text{inter}}$ grows with F. As is evident from Fig. 7, the larger the value of F, the lower the pre-FEC BER; hence, a larger throughput. Increasing the load has two opposing effects on $\mathcal{T}_{\text{inter}}$. First, with higher loads, more connections are set up, translating to higher throughput. Second, the BER increases with load (see Fig. 7), and consequently the code rate and throughput decrease. With 2-PAM and 4-PAM, the first effect dominates and the throughput constantly increases with load. However, with 8-PAM, the second effect becomes dominant for $F = 1, 2$ under large enough loads.

Using higher-order modulations *i)* increases the number of transmitted bits per symbol and consequently the throughput, and *ii)* increases the pre-FEC BER, which in turn decreases the code rate and throughput. As depicted in Fig. 8, by moving from 2-PAM to 4-PAM, the throughput increases as the first effect dominates. However, comparing Fig. 8 (b) and Fig. 8 (c), one can note that for a given load, the throughput decreases by moving from 4-PAM to 8-PAM. Therefore, 4 is the best modulation order to be used by the interdomain traffic in the multi-FSR transmission scenario.

VI. Conclusion

To address the switching bottlenecks imposed by the fixed AWG routing pattern, we investigated the impact of AWG FSR periodicity on the performance of a distributed multicast switch architecture. We developed a general-purpose, analytical framework to estimate the BP in a multistage setting. In addition, we conducted Monte

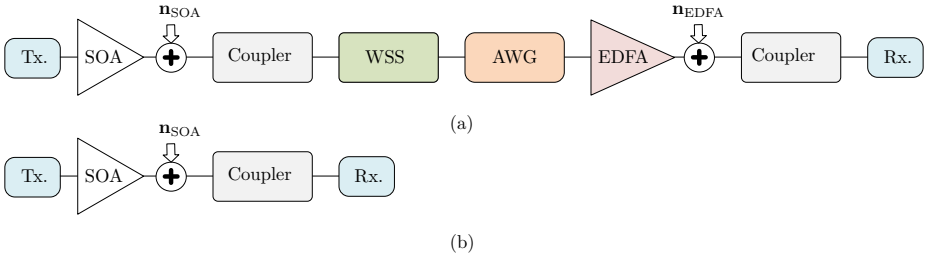

Fig. 6. Transmission path of (a) interdomain and (b) intradomain traffic for the switch in Fig. 1.

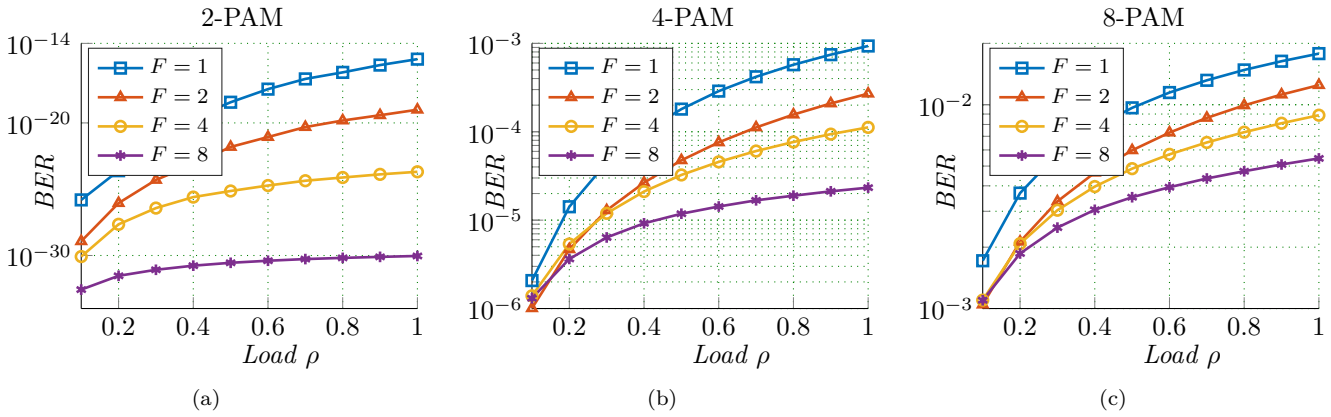

Fig. 7. BER for (a) 2-PAM, (b) 4-PAM, and (c) 8-PAM for $F = 1, 2, 4, 8$.

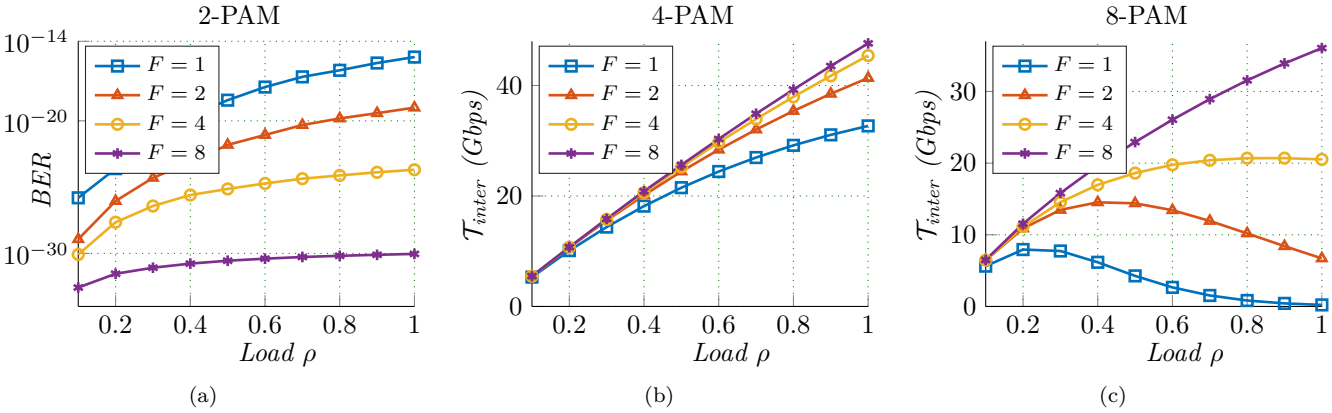

Fig. 8. $\mathcal{T}_{\text{inter}}$ (see (20)) for (a) 2-PAM, (b) 4-PAM, and (c) 8-PAM for $F = 1, 2, 4, 8$.

Carlo simulations to study the performance under different values of the FSR count. Considering the parameters of our study, the interdomain BP could be significantly improved by increasing the FSR count from one to four, with larger values resulting in diminishing returns and further reducing the number of supported nodes.

From a physical-layer standpoint, an increase in the FSR count leads to a decrease in BER and a larger effective bit rate per connection. In our cross-layer simulations, 4-PAM led to the highest normalized interdomain throughput for all of the considered FSR counts. In summary, the impact of the physical layer on wavelength-routing architectures can be minimized using multi-FSR solutions, low-crosstalk AWG devices, and adaptive coding and modulation formats. As the performance of the multistage switch is dependent on the traffic type, in the future, further investigation is needed to examine the switch performance under different traffic patterns. Another interesting extension would be to develop distributed scheduling algorithms for multicast traffic delivery in wavelength-routing switches.

Acknowledgment

This work was supported by the Swedish Research Council under grant no. 2014-6230, the NSF Center for Integrated Access Networks (CIAN) under grant no. EEC-0812072, and the Natural Sciences and Engineering Research Council of Canada (NSERC). The simulations were

carried out on the resources provided by the Swedish National Infrastructure for Computing (SNIC) at C3SE.

References

[1] P. K. Agyapong, M. Iwamura, D. Staehle, W. Kiess, and A. Benjebbour, "Design considerations for a 5G network architecture," *IEEE Communications Magazine*, vol. 52, no. 11, pp. 65–75, Nov. 2014.

[2] N. Wang, E. Hossain, and V. K. Bhargava, "Backhauling 5G small cells: a radio resource management perspective," *IEEE Wireless Communications*, vol. 22, no. 5, pp. 41–49, Oct. 2015.

[3] L. Velasco, A. Castro, A. Asensio, M. Ruiz, G. Liu, C. Qin, R. Proietti, and S. J. B. Yoo, "Meeting the requirements to deploy cloud RAN over optical networks," *Journal of Optical Communications and Networking*, vol. 9, no. 3, pp. B22–B32, Mar. 2017.

[4] M. R. Raza, M. Fiorani, A. Rostami, P. Öhlén, L. Wosinska, and P. Monti, "Demonstration of dynamic resource sharing benefits in an optical C-RAN," *Journal of Optical Communications and Networking*, vol. 8, no. 8, pp. 621–632, Aug. 2016.

[5] A. S. Gowda, L. G. Kazovsky, K. Wang, and D. Larrabeiti, "Quasi-passive optical infrastructure for future 5G wireless networks: pros and cons," *Journal of Optical Communications and Networking*, vol. 8, no. 12, pp. B111–B123, Dec. 2016.

[6] C. Kachris, K. Kanonakis, and I. Tomkos, "Optical interconnection networks in data centers: recent trends and future challenges," *IEEE Communications Magazine*, vol. 51, no. 9, pp. 39–45, Sep. 2013.

[7] J. Chen, Y. Gong, M. Fiorani, and S. Aleksic, "Optical interconnects at the top of the rack for energy-efficient data centers," *IEEE Communications Magazine*, vol. 53, no. 8, pp. 140–148, Aug. 2015.

[8] H. Rastegarfar, L. Yan, K. Szczerba, and E. Agrell, "PAM performance analysis in multicast-enabled wavelength-routing data centers," *Journal of Lightwave Technology*, vol. 35, no. 13, pp. 2569–2579, Jul. 2017.

[9] H. Rastegarfar, A. Leon-Garcia, S. LaRochelle, and L. A. Rusch, "Cross-layer performance analysis of recirculation buffers for optical data centers," *Journal of lightwave Technology*, vol. 31, no. 3, pp. 432–445, Feb. 2013.

[10] K.-I. Sato, H. Hasegawa, T. Niwa, and T. Watanabe, "A large-scale wavelength routing optical switch for data center networks," *IEEE Communications Magazine*, vol. 51, no. 9, pp. 46–52, Sep. 2013.

[11] K. Zhang, Q. Zhuge, H. Xin, H. He, W. Hu, and D. V. Plant, "Low-cost WDM fronthaul enabled by partitioned asymmetric AWGR with simultaneous flexible transceiver assignment and chirp management," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 9, no. 10, pp. 876–888, Oct. 2017.

[12] Q. Huang and W.-D. Zhong, "Wavelength-routed optical multicast packet switch with improved performance," *Journal of Lightwave Technology*, vol. 27, no. 24, pp. 5657–5664, Dec. 2009.

[13] M. Maier, M. Scheutzow, and M. Reisslein, "The arrayed-waveguide grating-based single-hop WDM network: an architecture for efficient multicasting," *IEEE Journal on Selected Areas in Communications*, vol. 21, no. 9, pp. 1414–1432, Nov. 2003.

[14] Z. Guo, J. Duan, and Y. Yang, "On-line multicast scheduling with bounded congestion in fat-tree data center networks," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 1, pp. 102–115, Jan. 2014.

[15] W.-K. Jia, "A scalable multicast source routing architecture for data center networks," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 1, pp. 116–123, Jan. 2014.

[16] D. Li, Y. Li, J. Wu, S. Su, and J. Yu, "ESM: Efficient and scalable data center multicast routing," *IEEE/ACM Transactions on Networking*, vol. 20, no. 3, pp. 944–955, Jun. 2012.

[17] D. Li, M. Xu, Y. Liu, X. Xie, Y. Cui, J. Wang, and G. Chen, "Reliable multicast in data center networks," *IEEE Transactions on Computers*, vol. 63, no. 8, pp. 2011–2024, Aug. 2014.

[18] H. Wang, Y. Xia, K. Bergman, T. S. Ng, S. Sahu, and K. Sripanidkulchai, "Rethinking the physical layer of data center networks of the next decade: Using optics to enable efficient *-cast connectivity," *ACM SIGCOMM Computer Communication Review*, vol. 43, no. 3, pp. 52–58, Jul. 2013.

[19] P. Samadi, V. Gupta, J. Xu, H. Wang, G. Zussman, and K. Bergman, "Optical multicast system for data center networks," *Optics Express*, vol. 23, no. 17, pp. 22 162–22 180, Aug. 2015.

[20] K. Keykhosravi, H. Rastegarfar, and E. Agrell, "Multicast scheduling of wavelength-tunable, multiqueue optical data center switches," *Journal of Optical Communications and Networking*, vol. 10, no. 4, pp. 353–364, Apr. 2018.

[21] W. Ni, C. Huang, Y. L. Liu, W. Li, K.-W. Leong, and J. Wu, "POXN: a new passive optical cross-connection network for low-cost power-efficient datacenters," *Journal of Lightwave Technology*, vol. 32, no. 8, pp. 1482–1500, Apr. 2014.

[22] T. Biermann, L. Scalia, C. Choi, W. Kellerer, and H. Karl, "How backhaul networks influence the feasibility of coordinated multipoint in cellular networks," *IEEE Communications Magazine*, vol. 51, no. 8, pp. 168–176, Aug. 2013.

[23] J. Zhang, Y. Ji, S. Jia, H. Li, X. Yu, and X. Wang, "Reconfigurable optical mobile fronthaul networks for coordinated multipoint transmission and reception in 5G," *Journal of Optical Communications and Networking*, vol. 9, no. 6, pp. 489–497, Jun. 2017.

[24] H. Rastegarfar, K. Keykhosravi, E. Agrell, and N. Peyghambarian, "Wavelength reuse for scalable multicasting: a cross-layer perspective," in *Optical Fiber Communication Conference*, Mar. 2018, p. W2A.20.

[25] C. Bock and J. Prat, "WDM/TDM PON experiments using the AWG free spectral range periodicity to transmit unicast and multicast data," *Optics Express*, vol. 13, no. 8, pp. 2887–2891, Apr. 2005.

[26] C. Bock, J. Prat, and S. D. Walker, "Hybrid WDM/TDM PON using the AWG FSR and featuring centralized light generation and dynamic bandwidth allocation," *J. Lightw. Technol.*, vol. 23, no. 12, pp. 3981–3988, Dec. 2005.

[27] Z. Xu, X. Cheng, Y.-K. Yeo, X. Shao, L. Zhou, and H. Zhang, "Large-scale WDM passive optical network based on cyclical AWG," *Optics Express*, vol. 20, no. 13, pp. 13 939–13 946, Jun. 2012.

[28] W. Feller, *An introduction to probability theory and its applications*, 3rd ed. Wiley, New York, 1968, vol. 1.

[29] S. Kamei, M. Ishii, A. Kaneko, T. Shibata, and M. Itoh, "$N \times N$ cyclic-frequency router with improved performance based on arrayed-waveguide grating," *J. Lightw. Technol.*, vol. 27, no. 18, pp. 4097–4104, Sep. 2009.

[30] WaveShaper 16000S, https://www.finisar.com/sites/default/files/downloads/waveshaper_16000s_product_brief_11_14_0.pdf, May 2019.

[31] Long-Reach DWDM SFP Transceiver, https://www.finisar.com/sites/default/files/downloads/fwlf1631r_long-reach_dwdm_sfp_transceiver_spec_revc1.pdf, Oct. 2019.

[32] $N \times N$ AWG Multiplexers and Demultiplexers Router Module, http://www.enablence.com/media/pdfs/Datasheet_OCSD_AWG_Other_NxN_APRTE_0.pdf, 2010.

[33] +/-800 ps/nm (40km) Tunable XFP Optical Transceiver, https://www.finisar.com/sites/default/files/downloads/finisar_ftlx6611mcc_xx_and_ftlx6614mcc_xx_-800_ps_nm_40km_tunable_xfp_optical_transceiver_product_specification_rev_b1.pdf, Oct. 2019.

[34] W. J. Ebel and W. H. Tranter, "The performance of Reed-Solomon codes on a bursty-noise channel," *IEEE Trans. Commun.*, vol. 43, no. 2/3/4, pp. 298–306, Feb.–Apr. 1995.